

Limitations of On Demand Secure Routing Protocols

^{**}Prabha Ramachandran, ^{§†}Alec Yasinsac (Contact Author)

Abstract – Routing in mobile ad hoc networks is an open and active area of research. Recently, many have attempted to add value to routing protocols by improving efficiency and security of these protocols. In this paper, we show that a whole class of routing protocols for dynamic networks is vulnerable to a subtle attack. We illustrate this attack on several well known protocols and describe the fundamental properties of this attack and of the protocols that are vulnerable to it. We also propose potential approaches to overcoming the vulnerability that we address.

Index terms – Mobile networks, routing, security, dynamic source routing, on-demand routing, invisible node attack

I. INTRODUCTION

Routing in Mobile Ad hoc Networks (MANETs) has received significant attention in the past five years [1]. Assuming limited capacity for state retention and dynamic structures that incur high overhead for maintenance, on demand protocols [2] are the paradigm of choice. In their seminal paper [3], Johnson, et al. introduce Dynamic Source Routing (DSR) followed by Perkins and Royer's Ad hoc On Demand Routing (AODV) [2]. These two protocols form the foundation of many subsequent attempts to create secure routing protocols, such as the Secure Routing Protocol (SRP) [4] and Ariadne [5]. Unfortunately, SRP is fundamentally flawed [6]. In this paper we show how the attack on SRP is applicable to the wide class of ALL protocols based on AODV and DSR.

In the past two years, research focus has shifted towards providing security in wireless networks. In their seminal work [7], Zhou and Haas recognize secure routing as an important issue in wireless communications. Because of its generally positive properties in the wireless

environment, AODV and DSR are natural choices as the foundation of many different protocols that add value by incorporating security mechanisms. For example, Papadimitratos and Haas [4] offer their *Secure Routing Protocol* (SRP). They make a compelling argument that their on demand routing protocol is secure against all attacks. The foundation of their claim is that end-to-end integrity protection combined with forward and reverse path matching prevents insertions or deletions from the discovered route. The fundamental error in their reasoning, pointed out by Marshall et al. [6] is that insertion and deletion of paths are not the only acts that can compromise the discovered route. By simply relaying a route request without modification, a malicious node can ensure that a false route is created.

In the following section, we show how the invisible node attack is applicable to several proposed secure routing protocols for ad hoc networks, and in section 3, we give an argument that all end-to-end, two pass route discovery protocols are vulnerable to the invisible node attack. We then describe the options for securing these protocols for routing networks and summarize and conclude the paper in section 5.

II. VULNERABILITIES IN SECURE ROUTING PROTOCOLS

AODV and DSR build a route through forward path discovery and confirm the route via reverse path reply. While each of these protocols retain state, since their primary focus is to provide “dynamic” and “on demand” routing, without loss of generality, we consider their properties relative to route discovery where no state exists. Attempts to secure these protocols focus on ensuring the inability of nodes to modify or delete input from previous nodes in the forward or reverse route. In this section, we demonstrate a chronic weakness in two pass, end-to-end protocols by showing weaknesses in various well-known attempts to create secure routing protocols for ad hoc networks based on AODV or DSR.

A. Secure Routing for Mobile Ad hoc Networks

Marshall, et al [6] demonstrated the invisible node attack against SRP [4], and discussed its impact in detail [8]. The attack is accomplished by the malicious node simply acting as a relay for both the forward path route request and the reverse path route reply. This results in agreement by the source and destination on a route that is dependant on the malicious node, but does not reflect that

* Department of Electrical and Computer Engineering
University of Maryland, College Park, MD
prabha@isr.umd.edu

§ Computer Science Department
Florida State University
Tallahassee, FL 32306-4530
yasinsac@cs.fsu.edu

† Prepared through collaborative participation in the Collaborative Technology Alliance for Communications & Networks sponsored by the U.S. Army Research Laboratory under Cooperative Agreement DAAD19-01-2-0011

‡ This material is based upon work supported in part by the U.S. Army Research Laboratory and the U.S. Army Research Office under grant number DAAD19-02-1-0235

dependency. The invisible node attack is the foundation of the weaknesses that we illustrate in the following subsections.

B. Authenticated Routing for Ad hoc Networks(ARAN)

Dahill et al. [9] present a secure routing protocol for ad hoc networks called Authenticated Routing for Ad hoc Networks (ARAN). ARAN uses certificates for authentication and non-repudiation, but does not protect privacy for its protocol messages. ARAN as defined is shown in Figure 1. Although ARAN does not follow a DSR-type route discovery and response, the key point is that each node on the path from the source to the destination is required to strip off the previous hop's signature and certificate and append its signature and certificate before broadcasting the packet. Any routing protocol in which an intermediate node is required to add some fields to the incoming request (reply) packet is

Authenticated Route Discovery

A→*: [RDP, IP_X, cert_A, N_A, t]K_A
 B→*: [[RDP, IP_X, cert_A, N_A, t]K_A] K_B, cert_B
 C→*: [[RDP, IP_X, cert_A, N_A, t]K_A] K_C, cert_C
 D→*: [[RDP, IP_X, cert_A, N_A, t]K_A] K_D, cert_D

Authenticated Route Setup

X→D: [REP, IP_A, cert_X, N_A, t]K_X
 D→C: [[REP, IP_A, cert_X, N_A, t]K_X]K_D, cert_D
 C→B: [[REP, IP_A, cert_X, N_A, t]K_X]K_C, cert_C
 B→A: [[REP, IP_A, cert_X, N_A, t]K_X]K_B, cert_B

Notation

node A's private key
 K_{A+} node A's public key
 [d]_{K_A} Data d digitally signed by A
 cert_A A's certificate
 t timestamp
 e certification expiration time
 N_A nonce issued by A
 IP_A IP address of A
 RDP Route Discovery Packed ID
 REP Reply Packet ID
 T Trusted Certificate Server
 Cert_A = [IP_A, K_A, t, e]K_T

Figure 1. ARAN Secure Routing Protocol

vulnerable to the invisible node attack as explained below. The attack on ARAN can easily be shown possible during route discovery and setup by any intermediate node. In the above example, if node C rebroadcasts node B's packet during route discovery and again relays node D's packet during route setup, nodes B and D are ignorant of C's presence. When node C does not forward data packets sent along this route, link breaks that are detected

do not identify the nodes correctly, i.e. C is never in the picture from node B and node D's point of view. This simple relay attack thus demonstrates a major security flaw in ARAN.

C. Security-aware Ad-hoc Routing protocol (SAR)

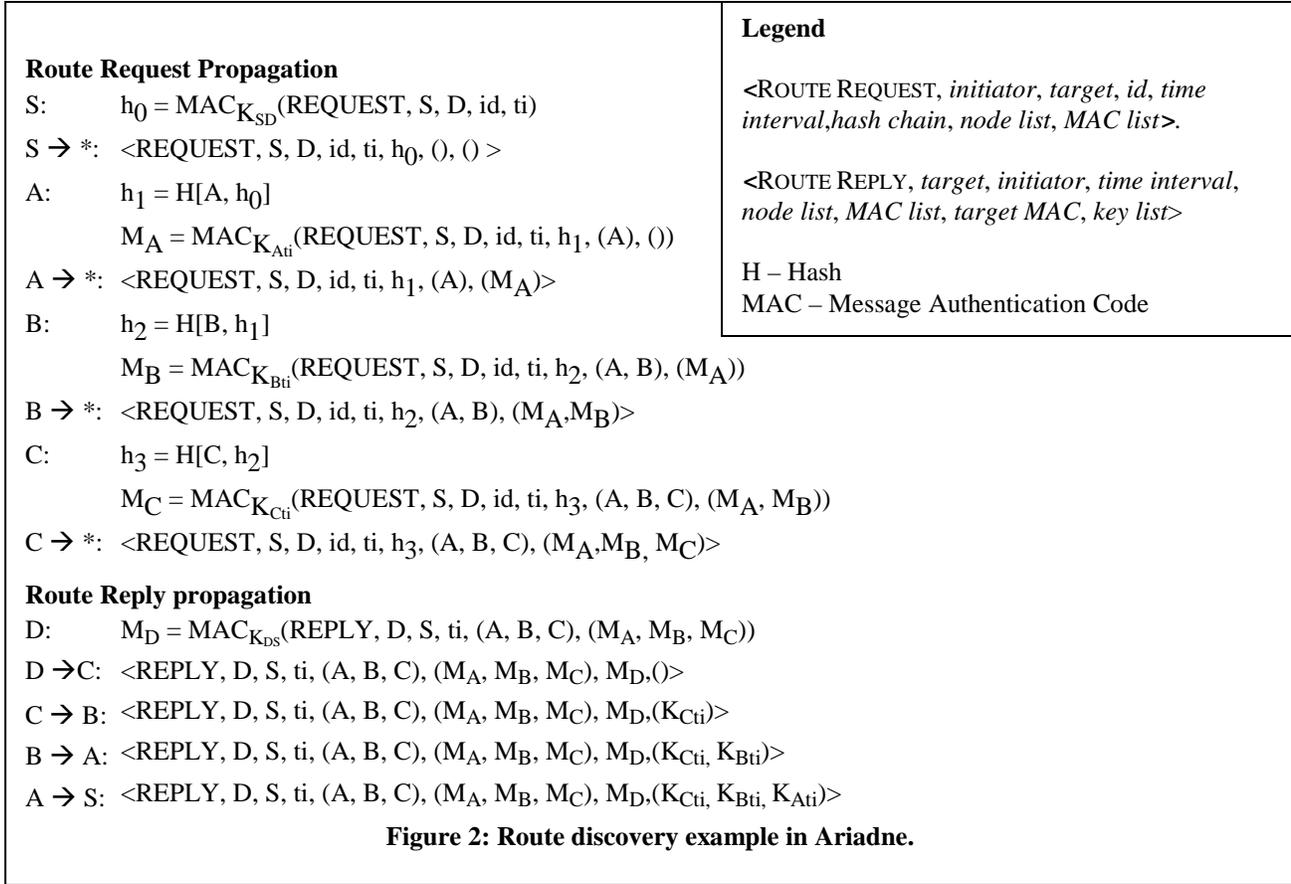
SAR [10] is a routing technique that embeds security metrics into the on-demand route discovery of any base protocol like AODV or DSR. The forwarding procedures are changed with respect to the Route REQuests (RREQ). The source node sets the (immutable) RQ_SEC_REQUIREMENT field in the RREQ and broadcasts the RREQ. On receipt of the RREQ packet, an intermediate node checks if it satisfies the security requirement indicated in the packet. If it does, it updates the RQ_SEC_GUARANTEE field before forwarding the packet. Otherwise, it drops the packet. Propagation of RREPs is similar to that of the base protocol.

Encrypting or digitally signing the RREQ headers will prevent tamper and fabrication of messages by a malicious node. However, if a malicious node M that does not satisfy the security requirement indicated in the RREQ rebroadcasts the incoming RREQ packet instead of dropping it as required by the protocol, two nodes A and B dependant on M to communicate with each other will now be fooled into believing that they are next-hop neighbors of each other. The arrival of the RREQ at the destination thus does not guarantee the presence of a path satisfying the security requirement specified by the sender. It is not clear from their work [10] as to how such malicious behavior can be thwarted.

D. Ariadne

Ariadne [5] is a promising protocol based on the novel broadcast authentication mechanism TESLA [11]. It uses a one way hash chain coupled with a MAC using a shared key between the initiator and target to authenticate the initiator at the target. Each intermediate node appends its MAC in addition to its address, thus enabling the initiator to authenticate each entry in the path in the route reply. Ariadne guarantees that no intermediate node can remove a previous node in the node list in the Request or Reply.

Figure 2 shows a simple route discovery example in Ariadne, following the notation used by the authors [5]. Assume nodes A and C are not in direct wireless transmission range and that node B can hear both node A and node C. A malicious node on the discovered route can make the source and the destination believe that the false route is a valid one. For instance, in the example shown in figure 2, if node B does not append its address and MAC to the incoming request packet from node A and simply



retransmits A's packet, node C considers node A as its next-hop entry for the reverse route. When node C forwards its Reply to node A, node B hears it and retransmits the packet making node A believe that the packet arrived from node C. Thus a lone malicious node on the path from the source to the destination makes every node on the route believe that {S,A,C,D} is a valid route. Capkun and Hubaux propose a secure routing scheme called BISS, that builds secure routes out of an incomplete set of Security Associations [12]. From a security point of view, it is as secure (and insecure) as Ariadne and hence is vulnerable to the invisible node attack.

E. Packet Leashes

Hu, et al. [13] propose a mechanism to link messages in on-demand, route discovery protocols. Packet leashes rely on either geographic or temporal mechanisms to connect the messages in the routing protocol with the goal of preventing either malicious insertions or deletions. Unfortunately, both mechanisms are vulnerable to the invisible node attack. We first address geographic leashes by providing a simple attack example.

1. An Attack on Geographical Leashes

The attack that we outline requires collaboration by two malicious nodes that are adjacent in the route. Assume source node S wants to discover a route to destination node D and the most advantageous route must relay through malicious nodes M_1 and M_2 . The following are the attack steps:

- (1) S forwards the RREQ, noting D as the destination
- (2) For the route request, M_1 relays the RREQ to M_2 without modification.
- (3) M_2 acts as a normal node, following the protocol to forward the packet to the destination D, providing all required location information.
- D now believes there is a legitimate route {S, M_2 , D} and prepares the RREP accordingly. On the return route, the actions are slightly different.
- (4) D forwards the RREP via unicast to M_2 .
- (5) M_2 relays the unchanged route reply to M_1 .
- (6) M_1 assumes M_2 's identity to compute and relay the route reply to S, including the correct location information as required in the protocol.

S now believes there is a legitimate route {S, M_2 , D}.

2. Vulnerabilities in Temporal Leashes

We now turn our attention to temporal leashes. We first present a single-intruder, invisible node attack and then a collusion attack. We conclude this subsection with some observations about packet leashes.

Single Intruder Attack on Temporal Leashes

The foundation of temporal leashes is that message delivery times are predictable, consistent, and measurable. The authors argue that estimating the propagation delay between two nodes and including information in the transmitted packets allows the receiving node to ensure that the packet has not propagated beyond the allowable time/distance. This is a very strong assumption. First, it is unreasonable to assume that a sophisticated intruder is not capable of accelerated transmission speed, either through higher bandwidth or compression. In addition, if transmission speeds are known, it is unreasonable to assume sufficient precision in transmission characteristics or the timing scheme to thwart the invisible node attack in the general case.

Assume that S desires to discover a route to D and the best route traverses only malicious node M . Consider the case where node M is near the outer limit of the transmission range of S , and further assume that node D is close to M , but beyond the transmission range of S . By the packet leash assumption, the sending/receiving time is negligible, so relay by the invisible node is undetectable in and of itself. Thus, the security of the packet leash algorithm is dependent upon precise knowledge of the maximum propagation delay, where even minimal variance would allow this attack. We contend that there is no known method to provide such precision, and that such a solution is highly unlikely, particularly in ad hoc networks that are routinely heterogeneous.

This leaves out-and-back protocols that employ packet leashes vulnerable to the invisible node attack as described above, with the caveat that the malicious node must arrange their proximity just inside the range of the intended victim node.

A Collusion Attack on Temporal Leashes

In addition to the single intruder attack, temporal leashes are also vulnerable to a collusion-oriented invisible node attack, similar to the attack against geographical packet leashes. As before, the attack requires collaboration by two malicious nodes that are adjacent in the desired route. Assume source node S wants to discover a route to destination node D and that the most advantageous route must relay through malicious nodes M_1 and M_2 . The following are the attack steps:

(1) S forwards the RREQ, noting D as the destination

(2) For the route request, M_1 relays the RREQ to M_2 without modification.

(3) M_2 acts as a normal node, following the protocol to forward the packet to the destination D , providing all required temporal information.

D now believes there is a legitimate route $\{S, M_2, D\}$ and prepares the RREP accordingly. On the return route, the actions mirror the earlier collusion attack.

(4) D forwards the RREP via unicast to M_2 .

(5) M_2 relays route reply to M_1 without modification.

(6) M_1 assumes M_2 's identity to compute and relay the route reply to S , including the correct location information as required in the protocol.

S now believes there is a legitimate route $\{S, M_2, D\}$ and the invisible node attack is complete.

3. Packet Leashes Summary

We have shown how packet leashes are vulnerable to the invisible node attack by single intruders and through collusion of two malicious nodes. While packet leashes may be modified to provide end-to-end temporal protection, this entails composition of link variances and compounds the inevitable timing and communication unpredictability in addition to the strong assumption of restricting intruder transmission and compression capabilities.

Attacks can be very difficult to envision and overconfidence is normal. The security analysis of Hu et al. [13] (Section VI.B.) is not a formal proof. Rather, it is an argument that relates to two specific attacks, the wormhole attack (where malicious nodes have special capabilities) and an attack with a malicious sender. It does not address all or even broad classes of attacks and cannot address the invisible node vulnerabilities where the source and destination nodes are both trusted and neither engages in malicious acts. Thus, packet leashes cannot be considered a general or comprehensive security approach for ad hoc networks.

4. Watchdog and Bloodhound

Packet leashes utilize location information or propagation delay to bind messages to one-hop neighbors. Others leverage the bi-directional nature of wireless devices to detect replay attacks. Watchdog [14], requires that each node listen for instances of proper message forwarding by its neighbors. When neighbors are detected as not properly forwarding messages, they are tagged as malicious. Bloodhound [6], conversely, requires that each node listen for exact replicas of its transmitted messages. If replicas are detected, the replay attack is identified and the route discovery process is aborted.

Unfortunately, each of these mechanisms poses a risk of denial of service and each can be thwarted by directional

antenna (pointed toward the transmitting node for watchdog, away from the transmitting node for bloodhound). As with packet leashes, Watchdog and Bloodhound are suitable in some situations, but neither they, nor packet leashes can be considered a comprehensive security solution for ad hoc networks.

5. Secure Position Aided Ad hoc Routing (SPAAR)

Position-based protocols [15,16,17] have been proposed to improve efficiency of MANET routing. Carter et al. [8] leverage the positive properties of position-based routing protocols to mitigate the invisible node attack and provide a routing protocol that is secure even in a hostile environment. Unfortunately, SPAAR neither claims nor attempts to prevent attacks by compromised or traitor nodes. The protocol specifically requires that all trusted nodes act in accordance with the protocol rules. Certainly, this is reasonable and usable in many environments. However, it also cannot be considered a comprehensive solution.

III. END-TO-END, TWO-PASS ROUTE DISCOVERY IS INHERENTLY INSECURE

While secure routing protocols for ad hoc networks are becoming common in the literature, proofs of security for these protocols are rare. The common approach taken [5, 13] is to present an informal security analysis of the proposed protocols. This is easy to understand because proofs of security are difficult in most environments. Fortunately, our claim in this paper is insecurity, and we argue that end-to-end, two pass route discovery protocols are an inherently insecure foundation for routing protocols in wireless, mobile networks.

Our first step is to identify characteristics that are fundamental to on-demand routing protocols like AODV and DSR. Certainly, any protocol has many different characteristics, some fundamental, others incidental to its operation. For AODV, we believe the structure that defines it is the request, reply rules as shown in Figure 3. Varieties of AODV and DSR have sought to improve efficiency by reducing the percentage of nodes transmitting the route request. Many researchers attempt to provide integrity of the established route and others protect the privacy of the routing messages.

The common characteristic that we are concerned with is the relationship between the route reply and the route request. A fundamental assumption that on demand, end-to-end protocols such as AODV and DSR rely upon is that forcing the route reply to be passed along the return route is not merely a simple and efficient solution; it also provides a foundation for ensuring the security and integrity of the discovered route. For example, the SRP

authors assume that because the route is authenticated at the destination, then only two attacks are possible:

- (1) If the route is maliciously modified during the request phase, the reply will not reach the origin
- (2) If the route is maliciously modified during the reply phase, the change will be detected by the origin

The invisible node attack illustrates the fallacy of this

- The route request is broadcast by the source
- The request is relayed, via broadcast, once by each intermediate node
- Upon receiving the route request, the destination returns the route via logical unicast, to the previous hop
- The reply is returned to the origin via the reverse route, by sequential logical unicast.

Figure 3. End-to-end Two Pass Route Discovery

assumption by showing that the discovered route need not be altered in either direction in order for the protocol to discover a false route. Specifically, the route request and reply may traverse links not reflected in the discovered route. Thus, secure protocols must rely on other mechanisms to protect the desired properties of route discovery.

The essence of the invisible node attack is that messages are constructed of bits that need not be modified when they are relayed. Thus, the bit stream itself CANNOT provide information about any intermediate relaying nodes. Temporal leashes reason about the propagation time of messages, while geographic leashes try to fix the proximity of transmitting nodes. Both approaches are legitimate, but have inherent weaknesses, as we showed earlier. In the following section, we propose geographic and temporal solutions, as well as addressing link level mechanisms.

IV. FIXING AD HOC ON DEMAND ROUTING PROTOCOLS

While the protocols that we investigate above are inherently insecure, it is possible to devise mechanisms that mitigate the vulnerabilities. To find these mechanisms, we must again examine the fundamental nature of on demand, end-to-end protocols. While AODV was developed for dynamic networks, it is important to understand that table-driven routing in general is not well-suited to dynamic networks. The purpose of routing algorithms is to reduce the overhead of route discovery by keeping route state at each node. If routes change too frequently, the advantage gained by keeping the state is overcome by the drain of routing errors and restarts [18]. Thus, we can reasonably assume that the rate of change is low in networks that employ routing protocols. This means that nodes either do not move very fast, or if they

move fast when they move, they do not move very often. We leverage this assumption by employing advanced, more resource intensive mechanisms for independent verification of the location of neighbor nodes. Using location information, we give approaches to resolving the invisible node attack.

A. Triangularization and Collaborative Tracing

In our approach to preventing the invisible node attack, we require each node to identify its location in each message. Of course, this is only useful if the receiving node can independently verify the location of the transmitting node. Fortunately, mechanisms for triangularization can allow nodes to collaboratively compute the precise location of originating nodes as long as another (trustworthy) neighbor receives the same signal. Our assumptions are:

- (1) Nodes can determine their own location
- (2) Nodes can detect the direction of received signals

Consider the ad hoc path SID between the source node S and destination node D through the intermediate node I shown in Figure 4. By assumption, nodes I and partner P know their own positions and the direction of the signal from S. P forwards its own position and the direction of the signal to node I. Node I computes angle SIP and sides SI and SP. Any of several trigonometric computations yield the position of S.

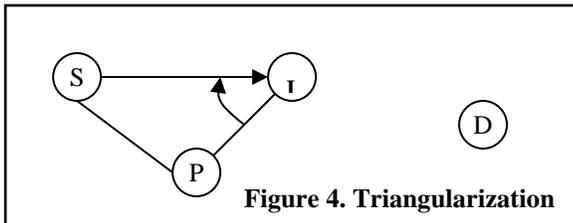


Figure 4. Triangularization

These calculations are not resource intensive, though they slightly increase the communication overhead by requiring one short location generation message between the receiving node and its partner. However, if the location information changes infrequently, the overhead may be reduced by storing location information of neighbors.

While collaborative tracing is vulnerable to an untrusted party attack, where P collaborates with S to fool I, the mechanism can overcome collaborative attacks in highly sensitive environments where high overhead is justified, using threshold techniques [19].

B. Computational Time Mandates

A critical assumption in the invisible node attack is that it is possible to relay a message without leaving a

significant sign or indicator that the message was relayed. Intruders leverage the fact that propagation delay is very short in ad hoc networks (6 μ sec/km [20]), making message relay difficult to detect via timing mechanisms. This is further complicated in ad hoc networks where synchronization to sufficient granularity is unlikely.

An approach to mitigating short propagation delay is to require that any relaying node perform a nontrivial, time predictable computation to guarantee a measurable time delay per hop. If each node is guaranteed to perform the computation, message relay is guaranteed to be detected.

The question then becomes how best to guarantee that nodes accomplish the computation before transmitting the message. We propose to use the time delay computation as a trap for intruders, where they can either elect to examine messages via some nontrivial processing before forwarding, or risk discovery of their malicious activity.

To accomplish this, we employ time-lock puzzles [21] embedded in each message. Time-lock puzzles allow a message to be kept secret for a minimum period of time (seconds, minutes, days, years, etc.) by employing mathematical formulas with well-understood computational time limits, e.g., by computing repetitive, sequential squaring. These puzzles are easy to derive and apply to the plaintext. In our case, the plaintext is the answer to the puzzle that indicates whether the message should be forwarded or not ("valid" or "dummy"). Dummy messages are generated occasionally with the specific purpose of detecting intruders. Malicious forwarding of dummy messages can be detected by the originator or by intermediate recipients. After sending a dummy message, the originator listens for a retransmission, and any retransmission of a dummy message indicates a protocol violation. This is similar to Bloodhound and Watchdog in that the originator listens for retransmissions. It differs in that only occasional dummy messages (not all messages) need be monitored.

As described above, originator listening can be defeated with directional antennas. However, if a receiving node detects a second iteration of a dummy message, it can also recognize that the second message was malicious. Intermediate node detection is much more difficult to avoid with directional antenna than is originator detection. In either case, when retransmission of a dummy message is detected, the detecting node can send a notification that a malicious node is in the vicinity or can begin malicious node identification as described above.

An algorithm for the time mandate mechanism is given in Figure 5. We recognize that computational time mandates inject significant overhead on the nodes and provide a target for denial of service. As always, security mechanisms are subject to competing objectives. We

contend that mitigating the invisible node attack will require such sacrifices.

- (1) The originating node generates an appropriate (valid or dummy) time-delay computation, an authenticated time stamp, and an authenticated hop counter, binds them to the message, and transmits the message. If a dummy message was transmitted, listen for retransmission and take remedial action if so detected.
- (2) Intermediate nodes solve the computation. If the message is a dummy, discard the message, listen for retransmission of the same message, and if detected, remediate the detected malicious activity. If the puzzle indicates that the received message is valid, check the timestamp and hop count. If the difference between the present time and the timestamp is greater than the total hop delay (hop delay threshold multiplied by the hop count), signal failure. Otherwise, increment the hop count, generate a new time-delay computation, bind them to the message, and retransmit.
- (3) The destination node accomplishes the same checks as intermediate nodes and performs the same activities when failure or malicious activity is detected. If the message is valid and the timestamp and hop count are consistent, the destination node generates and transmits the route reply.
- (4) The route reply proceeds as normal. No time mandates are necessary for the route reply since the invisible node attack requires invisible relay during the route request process.

Figure 5. Computational Time Mandates

C. Link-level Mechanisms

One proposed approach for protecting ad hoc networks is to establish strong message authentication between neighbors [8]. Conceptually, this may be considered a link level function. Unfortunately, the invisible node attack is not resolved by strong neighbor authentication, since authentication only indicates the identity of the *originator* of a message. To prevent the invisible node attack, it is necessary to know with certainty the identity of the *transmitter* of a message.

In essence, the invisible node attack is a variation of the classic man in the middle attack. Desmedt et al. [22] give two general solutions for resolving the man in the middle attack: temporal and geographic. These approaches are

repeated by Hu et al. [13]. It is unlikely that mechanisms at the link level can resolve this problem more efficiently or more effectively than a network layer protocol.

Additionally, even if physical or link layer mechanisms for transmission authentication can be found, effective security protocols are notoriously difficult to construct and verify. Protocol composition is even more difficult. Combining two secure protocols can easily result in subtle flaws that compromise the security of all composed protocols. This presents inherent danger in attempting to share responsibilities between link and network layers. While security protocol composition is an active research area [23, 24, 25], the principle of *caveat emptor* still applies; that is, if a function is necessary for the security of the protocol, the protocol must either accomplish the function itself, or verify that the function was accomplished properly. Mixing link and network protocols in this way injects complexity and systemic insecurity into the system.

V. CONCLUSIONS

In this paper we analyze a class of protocols proposed for ad hoc networks and show that any out-and-back route discovery protocol in which intermediate nodes fix the reverse path during the forward (request) path is vulnerable to the invisible node attack. Sophisticated mechanisms are required to counter or mitigate these vulnerabilities and we discuss some approaches to addressing this threat.

We have further shown that several important routing protocols previously thought to be secure are not secure and we illustrate the vulnerabilities with specific, reasonable attacks. We further argue that the fundamental properties of proposed routing protocols for dynamic and ad hoc networks merit close examination, and we use their properties to describe why an entire class of on demand routing protocols is vulnerable to attack.

Finally, we note that while finding routing mechanisms in ad hoc networks continues as an active research area, little work has been done to show that these mechanisms actually accomplish their goal. Additionally, where performance and efficiency arguments are made, important assumptions about network mobility, density, and heterogeneity characteristics are often omitted. In many cases, analytical argument is ignored and empirical evidence is all that is offered. We contend that such experimental data must be founded on theoretical grounds, and very little of that foundation presently exists for routing in ad hoc networks.

VI. REFERENCES

-
- [1] S. Corson and J. Macker, Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, RFC2501, January 1999.
- [2] C. Perkins and E. Royer, Ad hoc On-Demand Distance Vector Routing, *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, 1999, 90-100.
- [3] D. Johnson, D. Maltz, J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad Hoc Networks," *Ad Hoc Networking 2001*, pp. 139-172.
- [4] Papadimitratos and Z.J. Haas, "Secure Routing for Mobile Ad hoc Networks", *Proceedings of SCS Communication Networks and Distributed Systems Modeling and Simulation Conference*, San Antonio, USA, 2002.
- [5] Y-C. Hu A. Perrig, and D. Johnson, "Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks", in the 8th *ACM International Conference on Mobile Computing and Networking (MobiCom)*, September 2002.
- [6] J. Marshall, V. Thakur, and A. Yasinsac, "Identifying Flaws in the Secure Routing Protocol", *Proceedings of The 22nd International Performance, Computing, and Communications Conference*, April 9-11, 2003, pp. 167-174.
- [7] L. Zhou and Z.J. Haas, Securing Ad Hoc Networks, *IEEE Network Magazine*, vol. 13, no.6, Dec 1999
- [8] S. Carter and A. Yasinsac, "Secure Position Aided Ad hoc Routing Protocol", *Proceedings of the IASTED International Conference on Communications and Computer Networks (CCN02)*, pp. 329-334, Nov 4-7, 2002
- [9] B. Dahill, B. Levine, E. Royer, and C. Shields, "A Secure Routing Protocol for Ad Hoc Networks", Univ. of Massachusetts Tech. Rep. 01-37, 2001.
- [10] S. Yi, P. Nalburg and R. Kravets, Security-Aware Ad-Hoc Routing for Wireless Networks, UIUCDCS-R-2001-2241 Technical Report, 2001.
- [11] A. Perrig, R. Canetti, J. Tygar, and D. Song, "Efficient Authentication and Signing of Multicast Streams over Lossy Channels", *2000 IEEE Symposium on Security and Privacy*, pp. 56-73
- [12] S. Capkun and J-P. Hubaux, "BISS: Building Secure Routing Out of an Incomplete Set of Security Associations," *ACM WiSE 2003*.
- [13] Y. Hu, A. Perrig, D. Johnson, "Packet Leashes: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks", *Proceedings of The 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, March 2003.
- [14] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks", *Mobile Computing and Networking*, (2000), 255-65.
- [15] B. Karp and H. Kung, "Greedy Perimeter Stateless Routing for Wireless Networks", *Procs of the 6th International Conference on Mobile Computing and Networking*, Boston, MA, 2000, 243-54.
- [16] E. Royer and C-K. Toh, "A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks," *IEEE Personal Communications Magazine*, April 1999, 46-55.
- [17] Y. Ko, and N. Vaidya, Location Aided Routing in Mobile Ad Hoc Networks, *Proceedings of the 4th International Conference on Mobile Computing and Networking*, Dallas, USA, 1998.
- [18] A. Yasinsac, "Rate of Change in Ad Hoc Networks", *Proceedings of the Information Resources Management Association International Conference*, pp. 698-701, May 18-21, 2003, Philadelphia, PA.
- [19] Y. G. Desmedt, "Threshold cryptography," *European Trans. on Telecommunications*, 5(4), pp. 449-457, July-August 1994
- [20] Andrew S. Tannenbaum, Computer Networks, Prentice Hall, 1981, ISBN 0-13-165183-8, p. 113
- [21] R. L. Rivest, A. Shamir, and D. Wagner, "Time-loc Puzzles and Timed-release Crypto", Technical Report MIT/LCS/TR-684, 1996.
- [22] T. Beth & Y. Desmedt, "Identification Tokens-or: Solving the Chess Grandmaster Problem", In A. J. Menezes & S.A. Vanstone, editors *Proc. CRYPTO 90*, pp. 169-77. Springer-Verlag, '91. LNCS, 537.
- [23] R. Canetti, Y. Lindell, R. Ostrovsky, and A. Sahai. Universally composable two-party and multy-party secure computation. In *34th ACM Symposium on Theory of Computing*, pages 484-503, 2002
- [24] P. Mateus, J. Mitchell, and A. Secdrov, "Composition of Cryptographic Protocols in a Probabilistic Polynomial-Time Process Calculus", In: R. Amadio and D. Lugiez, eds., "CONCUR 2003 - Concurrency Theory, 14-th International Conference, Marseille, France, Sept. 2003", LNCS Volume 2761, Springer-Verlag, 2003, pp. 327-349
- [25] Hassen Saidi, Victoria Stavridou, and Bruno Duterte, "Protocol Codesign", *Cambridge International Workshop on Security Protocols*, 2-4 April 2003 (Pre-proceedings).